

Bases de données

Les outils informatiques permettent de conserver de grandes quantités de *données* que l’on peut manipuler pour en extraire de l’information. *L’ensemble des données rattachées à un thème particulier et stockées sur une machine capable de les traiter, constitue une base de données.* Ces données sont généralement liées les unes aux autres par des liens logiques traduisant les spécificités du thème concerné.

Depuis les années 1960, les informaticiens ont développé diverses manières de structurer ces données et d’interagir avec elles. À partir de années 70 et 80, une organisation logique particulière, appelée *modèle relationnel*, s’est révélée plus efficace que ses concurrentes et a été largement adoptée. Conformément au programme, ce sont ces *bases de données relationnelles (BDDR)* que nous allons étudier. Dans ce chapitre, nous en donnons les fondements conceptuels alors que leur interrogation sur machine, au travers du langage SQL, fera l’objet du chapitre suivant.

I Exemple introductif

I.1 À la confiserie

Madame Tartine, patronne de la confiserie « Les Délices de Nancy », se préoccupe de la bonne marche de ses affaires. Avant de prendre de stratégiques décisions de gestion, elle a décidé de recueillir des informations concernant les transactions qui ont lieu dans sa boutique et sur son site web. À chaque fois qu’un client passe en caisse, un employé inscrit sur une feuille les informations suivantes :

- nom et prénom du client ;
- adresse postale et adresse électronique du client ;
- pour chaque article acheté, son appellation, son prix, le nombre que le client en a pris et le montant total associé ;
- date de l’achat ;
- le montant total des achats de ce client dans ce passage en caisse.

Après quelques semaines, ces informations occupent une multitude de pages et, malgré tous les efforts des secrétaires, Madame Tartine ne parvient pas à répondre à des questions aussi simples que : « quel est, parmi tous nos clients, celui mange le plus de nougats ? » ou encore « dans quelle commune mes bonbons sont-ils le plus consommés ? ».

Naturellement, l’utilisation de papier est archaïque et un employé débrouillard s’avise qu’il serait plus efficace d’enregistrer toutes les informations dans un fichier informatique. La figure 1 montre à quoi ressemblent les premières lignes que l’ingénieur employé inscrit dans un tableur.

I.2 Décomposition en plusieurs tableaux

Par rapport au papier, ce tableau constitue certainement un progrès, mais il présente encore bien des inconvénients.

- On n’a pas relevé le montant total des achats effectués par un client donné au cours d’un même passage en caisse. On pourrait inscrire cette information dans des lignes s’insérant entre les autres sous forme de sous-totaux, mais cela rendrait le tableau difficilement lisible puisque les lignes ne seraient plus toutes de même nature, et qu’une colonne donnée contiendrait tantôt la valeur d’un article, tantôt un sous-total.
- Un même client peut apparaître plusieurs fois et il est fastidieux de recopier à chaque fois les informations le concernant. C’est même une source possible d’erreur, comme on peut le voir avec l’orthographe incorrecte « Dombs ».
- Un même article peut apparaître plusieurs fois, ce qui entraîne à nouveau une perte de temps et un risque d’erreur, sur son prix par exemple.
- Si un client change d’adresse électronique, il faut modifier de multiples lignes du tableau.
- Il peut y avoir des erreurs de calcul du total à payer sur chaque ligne.
- L’adresse n’est pas toujours présentée de la même manière.

Ces considérations invitent à organiser l’information différemment, en créant plusieurs tableaux de manière à *éviter des redondances* (figure 2).

On remarque tout d’abord qu’on a placé dans chaque tableau de la figure (2) des informations concernant

Nom	Prénom	adresse	courriel	article	prix	nbre	tot	date
Dujardin	Jean	31 Av. Foch, Paris	jean.duj@gmail.com	dragées chocolat	0,1	50	5	3 janvier 2022
Mbapé	Kilian	25 av. du 8 mai, Boulogne	mbape.k@psg.fr	sucettes à la fraise	0,3	20	6	6 février 2022
Van Laeke	Angèle	1, rue Lenoir, Uccle	agl@angele.be	sucettes à la fraise	0,3	10	3	2 avril
Macron	Émmanuel	Palais de l'Ély- sée	president@gouv.fr	pralines	0,2	100	20	25/12/2022
Klein	Matthieu	Mairie de Nancy	mat.klein@nancy.fr	bourgeon de sapin	0,1	10	1	10 mars
Domps	André	33 rue Kléber 54000 Nancy	domps.andre@neuf.fr	marrons glacés	0,5	20	10	1/3/22
Dombs	André	33 rue Kléber 54000 Nancy	domps.andre@neuf.fr	dragées au chocolat	0,5	10	5	1/3/22
Dujardin	Jean	31 Av. Foch, Paris	jean.duj@gmail.com	nougats	0,1	20	2	14 avril 2022
Lagarde	Christine	12, Wursts- trasse, Frank- furt	c.lagarde@bce.fr	pralines	0,2	10	2	12/3
...

FIGURE 1 – Écriture en vrac, dans un unique tableau, des données de la confiserie

des entités de même nature (par exemple les clients). Cette présentation en quatre tableaux présente plusieurs avantages.

- On a supprimé les redondances : chaque client apparaît une fois et une seule. Chaque article apparaît une fois et une seule. On réalise ainsi une économie d’espace mémoire et il devient aisé de modifier un élément tel que l’adresse d’un client ou le prix d’un article.
- On peut facilement ajouter de l’information, par exemple le pays de résidence de chaque client. Il suffit pour cela de créer une nouvelle colonne dans le tableau **CLIENT**.
- On a supprimé les données qu’on peut calculer à partir des autres (prix total pour un article dans un achat donné).
- On a uniformisé l’écriture des adresses et des dates.
- Par quelques opérations, on peut obtenir le montant total des achats d’un client lors d’un même passage en caisse, alors que cette information était indisponible dans le grand tableau fourre-tout initial.

Voyons comment on utilise ces tableaux. Considérons la commande numéro 6361. Elle a été passée par le client numéro 5, et le tableau **CLIENT** nous dit qu’il s’agit d’André DOMPS. Dans la table **ACHAT**, on voit que cette commande comporte 20 fois l’article numéro 4 et 10 fois l’article numéro 1. Grâce à la table **ARTICLE**, nous voyons qu’il s’agit de marrons glacés et de dragées au chocolat, et nous calculons le montant payé par ce client par l’opération $20 \times 0,5 + 10 \times 0,1 = 11\text{€}$.

Répondre aux questions suivantes :

- a) Qu’a commandé Jean-Dujardin ?
- b) Quelle est l’adresse des clients ayant commandé des pralines ?
- c) Qui a acheté quoi le 6 février, et pour quel montant ?
- d) Les habitants de Nancy ont-ils dépensé plus ou moins d’argent que ceux de Paris ?

On voit que cette présentation de l’information en quatre tableaux permet de répondre à toutes ces questions. Pour des tableaux comportant des milliers de lignes, il deviendrait fastidieux de faire ce travail à la main, mais nous verrons dans le chapitre suivant qu’un ordinateur peut le faire à notre place au travers de requêtes formulées en langage SQL. Avec cet outil, nous pourrions exploiter efficacement l’information contenue dans des fichiers contenant des mégaoctets de données. Le patron de la confiserie pourrait ainsi répondre à des questions plus complexes que les précédentes, par exemple

- e) Dans quelle ville mange-t-on le plus de nougats ?

CLIENTS							
num_client	Nom	Prénom	Ville	Code post.	Voie	n°	courriel
0	Van Laeke	Angèle	Uccle	1180	rue Lenoir	1	agl@angele.be
1	Dujardin	Jean	Paris	75001	Av. Foch	31	jean.duj@gmail.com
6	Mbapé	Kilian	Boulogne	92100	Av. du 8 mai	25	mbape.k@psg.fr
3	Klein	Matthieu	Nancy	54000	place Stanislas	1	mat.klein@nancy.fr
4	Macron	Émmanuel	Paris	75008	rue du Faubourg- Saint-Honoré	55	president@gouv.fr
5	Domps	André	Nancy	54000	rue Kléber	33	domps.andre@neuf.fr
2	Lagarde	Christine	Frankfurt	60311	Wurststrasse	12	c.lagarde@bce.fr

ARTICLE			COMMANDE			ACHAT		
id_art	nom art	prix	n° commande	num_client	date	commande	id_art	nbr
0	nougat	0,1	3977	1	3/1/2022	3977	1	50
1	dragées au chocolat	0,1	4023	6	6/2/2022	4023	2	20
2	sucettes à la fraise	0,3	1987	4	25/12/2022	1987	5	100
3	bourgeon de sapin	0,1	1098	3	10/03/2022	1098	3	10
4	marron glacé	0,5	6361	5	1/3/2022	6361	4	20
5	praline	0,2	7621	1	14/04/2022	6361	1	10
			2987	0	2/4/2022	7621	0	20
			1654	2	12/3/2022	2987	2	10
						1654	5	10

FIGURE 2 – Données de la confiserie décomposées en 4 tableaux

- f) Les habitants de Nancy qui habitent un numéro pair mangent plus de pralines que les habitants de Bruxelles mangent de marrons glacés ?
- g) Les confiseries contenant du sirop de glucose produisent-elles plus de chiffre d’affaire que celles qui contiennent du chocolat ?

Avec l’information disponible à ce stade, nous ne pouvons pas répondre à la dernière question. Il faudrait pour cela créer un nouveau tableau renseignant sur les ingrédients intervenant dans la fabrication de chaque article. Les tableaux déjà existant conserveraient cependant leur pertinence et il ne serait pas nécessaire de les modifier. C’est un nouvel avantage de l’utilisation de tables séparées en lieu et place du stockage de l’information dans un unique tableau.

II Bases de données relationnelles

L’exemple traité dans la partie précédente a été l’occasion de rencontrer les principales caractéristiques des *bases de données relationnelles (BDDR)*. Cette partie constitue une généralisation et définit les notions prévues au programme de CPGE.

II.1 Concepts et vocabulaire

- Une *base de données relationnelle* organise l’information sous la forme d’un ensemble de tableaux, que l’on désigne plutôt sous le nom de *tables* (ou *relations*). Le mot « relation » est souvent utilisé dans le contexte plus abstrait de l’algèbre relationnelle alors que le mot « table » désigne plutôt la représentation concrète d’une relation, mais à notre niveau nous pouvons les confondre.
- Une table regroupe des informations concernant des *entités* de même nature. Une *entité* est un objet (concret ou abstrait) ou un fait du monde réel (hors de l’ordinateur) au sujet duquel on souhaite stocker des informations.
- On appelle *attribut* une propriété attachée à une entité. Le *domaine d’un attribut* est l’ensemble dans lequel un attribut peut prendre ses valeurs. Il lui correspond informatiquement un *type* de données, par exemple le type entier, le type flottant ou le type chaîne¹.

1. Il existe en SQL d’autres types que nous rencontrerons occasionnellement, comme le type DATE, mais leur connaissance ne figure

- Une ligne est la suite des valeurs prises par les différents attributs d'une même entité.
- Une table est un ensemble de *lignes* (ou *enregistrements*). L'affichage de ces lignes sur un écran peut laisser penser que celle qui apparaît tout en haut est la « première » et celle qui apparaît tout en bas est la dernière. En réalité, ces lignes ne sont pas ordonnées, conformément au concept mathématique d'ensemble. Les lignes d'une table sont distinctes.
- Dans une table donnée, les valeurs prises par un même attribut pour les différentes entités forment une *colonne*. On désigne cette colonne par le nom de l'attribut en question, ce qui, par un léger abus de langage, rend presque synonymes les mots « colonne » et « attribut ».
- Soient A_1, A_2, \dots, A_n les attributs d'une table donnée et soient $\mathcal{D}_1, \mathcal{D}_2, \dots, \mathcal{D}_n$ les domaines associés. Cette table est une partie du produit cartésien $\mathcal{D}_1 \times \mathcal{D}_2 \times \dots \times \mathcal{D}_n$, c'est à dire un ensemble de n-uplets $t = (t_1, t_2, \dots, t_n)$ avec $t_1 \in \mathcal{D}_1, t_2 \in \mathcal{D}_2, \dots, t_n \in \mathcal{D}_n$.
- Une table T est définie par son *schéma de table* (ou schéma de relation). C'est l'expression $T(A_1 : \mathcal{D}_1, A_2 : \mathcal{D}_2, \dots, A_n : \mathcal{D}_n)$ qui énumère chacun de ses attributs et leur domaine.

Appliquons ce vocabulaire à l'exemple de la partie I. La base de données CONFISERIE est formée de quatre tables dont voici les schémas de relation.

— La table CLIENT de schéma

CLIENT(numero : entier, Nom : chaîne, Prénom : Chaîne, Ville : Chaîne, Code post. : entier, Voie : chaîne, numéro : entier, courriel : chaîne). Ses entités sont les clients et chaque client possède 8 attributs, de sorte que la table possède 8 colonnes. L'extrait de table fourni comporte 7 enregistrements.

— La table ARTICLE de schéma

ARTICLE(id_art : entier, nom art : chaîne, prix : flottant). Ses entités sont les articles et chaque article possède 3 attributs : son identifiant, son nom et son prix, de sorte que la table possède 3 colonnes. L'extrait de table fourni comporte 6 enregistrements.

— La table COMMANDE de schéma

COMMANDE(numéro de commande : entier, numéro de client : entier, date : DATE). Ses entités sont les commandes et chaque commande possède 3 attributs : son numéro, le numéro du client qui l'a passée, sa date, de sorte que la table possède 3 colonnes. L'extrait de table fourni comporte 8 enregistrements.

— La table ACHAT de schéma

ACHAT(commande : entier, id_art : entier, nombre : entier). Ses entités sont les achats et chaque achat possède trois attributs : le numéro de commande au cours de laquelle l'achat a été effectué, l'identifiant d'un article, le nombre de tels articles achetés, de sorte que la table possède 3 colonnes. L'extrait de table fourni possède 9 enregistrements.

II.2 Ne pas perdre ses clés !

Les bases de données relationnelles reposent sur la décomposition de l'information en un certain nombre de tables, qui se substituent avantageusement à un unique tableau qui regrouperait toutes les données en vrac. Cette décomposition ne doit cependant pas faire disparaître les liens logiques qui existent entre des entités distinctes. Reprenons l'exemple de la question c) concernant la table CONFISERIE. Pour y répondre, nous cherchons d'abord dans la table COMMANDE : nous voyons que seule la commande 4023 convient et qu'elle concerne le client numéro 6. En cherchant ce numéro 6 dans la table CLIENT, nous voyons qu'il s'agit de K. Mbapé. Avec le 4023, nous voyons dans la table ACHAT qu'il a commandé 20 fois l'article numéro 2, et enfin cet entier 2 nous apprend qu'il s'agit de sucettes à la fraise coûtant 0,3€. Le numéro de client égal à 6, le numéro de commande 4023 et le numéro d'article valant 2 nous ont permis de passer d'une table à l'autre sans perdre le fil. Les attributs numéro de client, numéro de commande et id_art jouent donc un rôle particulier. On remarque d'ailleurs qu'ils n'existaient pas dans la présentation « brute » des données au paragraphe I.1. C'est précisément la décomposition en quatre tables qui a rendu nécessaire leur introduction. Nous allons préciser cette idée au travers du concept de *clé* (ou *d'identifiant*)².

- On appelle *clé* (ou *identifiant*) d'une table un ensemble d'attributs qui permet d'identifier une ligne sans ambiguïté. Autrement dit, il ne peut pas exister deux lignes pour lesquelles ces attributs possèdent les mêmes

pas au programme.

2. Le terme de « clé », traduction littérale de l'anglais « key », est le plus utilisé et choisi par le programme de CPGE. L'auteur de référence J.L. Hainaut privilégie celui d'identifiant

valeurs. Soit $C = \{A_{j_1}, A_{j_2}, \dots\}$ cet ensemble d’attributs et soient $t = (t_1, t_2, \dots, t_n)$ et $u = (u_1, u_2, \dots, u_n)$ deux lignes de la table. On a

$$t_{j_1} = u_{j_1}, t_{j_2} = u_{j_2}, \dots \Rightarrow t = u \quad .$$

Une clé peut être formée d’un seul attribut, mais il est parfois nécessaire d’en utiliser davantage. Pour la table `ARTICLE`, `id_art` et `nom art` sont des clés possibles. Pour la table `ACHAT`, `{numero de commande, id_art}` est une clé. Le numéro de commande seul ne suffit pas, comme le montre la valeur 6361. Je vous laisse trouver des clés pour les tables `CLIENTS` et `COMMANDE`.

- Parmi les diverses clés possibles d’une table donnée, on choisit d’en utiliser une en particulier et de délaissier les autres. La clé retenue se nomme *clé primaire*. Il est d’usage de souligner les noms des colonnes (des attributs) formant la clé primaire, notamment lorsqu’on écrit le schéma de la table. Pour la table `CLIENT` par exemple, `numero client` et `courriel` sont deux clés possibles, mais nous avons choisi `numero client` qui possède dès lors le statut de clé primaire.

Le choix d’une clé primaire permet d’indexer tous les enregistrements d’une table et facilite souvent sa manipulation.

```
CLIENT(numero client : entier, Nom : chaîne, Prénom : Chaîne, Ville : Chaîne,
Code post. : entier, Voie : chaîne, numéro : entier, courrier : chaîne)
ARTICLE(id_art : entier, nom art : chaîne, prix : flottant)
ACHAT(numéro de commande : entier, id_art : entier, nombre : entier)
```

- Dans une table donnée, une clé est dite dite *minimale* s’il n’en existe pas d’autre possédant un nombre d’attributs inférieur. Supprimer un seul attribut d’une clé minimale lui ferait perdre son rôle d’identifiant. Pour la table `ARTICLE` par exemple, l’ensemble d’attributs `id_art, nom art` constitue une clé, mais on peut se contenter de `id_art` : cet identifiant formé d’un unique attribut est minimal alors que le premier ne l’est pas. Pour la table `ACHAT`, les attributs `numero de commande, id_art` forment une clé minimale car la suppression de l’un quelconque d’entre eux introduirait une ambiguïté dans l’identification d’une ligne. En effet, deux lignes possèdent le numéro de commande 6361 et deux autres lignes vérifient `id_art = 1`. Bien évidemment, on choisit la clé primaire de telle sorte qu’elle soit minimale.
- Dans une table donnée, une *clé étrangère* est un ensemble d’attributs qui forment une clé pour une autre table, appelée *table cible*. En pratique, on convient de choisir comme clé étrangère la clé primaire de la table cible. Une clé étrangère permet de faire référence sans ambiguïté à des lignes de la table cible. Dans de rares cas, la première table (« table référençante ») peut être la même que la table cible (« table référencée »). On met parfois en valeur les attributs formant une clé étrangère en les faisant précéder d’un dièse, ou en les écrivant en italique, ou encore en les soulignant par une accolade. Cette pratique n’est pas universelle. Dans la table `COMMANDE`, la colonne `numero client` constitue une clé étrangère vers la table cible `CLIENT`. Dans la table `ACHAT`, la colonne `commande` est une clé étrangère vers la table cible `COMMANDE`. On remarque que le même attribut ne possède pas le même nom dans les deux tables. Cette situation se rencontre fréquemment et ne présente pas d’inconvénient.

II.3 Extension de la base de données CONFISERIE

Le patron de la confiserie s’enthousiasme de l’utilisation de sa base de données relationnelle, particulièrement depuis qu’il a acquis quelques rudiments de langage SQL pour l’interroger commodément. Il décide donc d’étendre la base `CONFISERIE` afin de mieux analyser les achats d’ingrédients qu’il réalise auprès de ses fournisseurs (sucre, farine, arômes, etc). Chaque recette de bonbon nécessite un certain nombre d’ingrédients, chacun en une quantité déterminée, et on peut les acheter chez divers fournisseurs qui proposent chacun un certain prix unitaire. Comment organiser cette information ? Je vous laisser réfléchir un peu avant de vous présenter la solution que j’ai retenue. On crée quatre nouvelles tables (figure 3).

- table `FOURNISSEUR`(id_four : entier, nom : chaîne, tel : entier)
Elle énumère les fournisseurs avec leur nom et leur numéro de téléphone et utilise comme clé primaire un entier `id_four`.
- table `OFFRE`(produit : chaîne, fournisseur : entier, prix : flottant)
Elle énumère les prix catalogues des différents produits (par kilogrammes pour ceux vendus au poids, par unité pour les autres), chez chacun des fournisseurs qui les proposent. Sa clé primaire est formée des attributs

{produit, fournisseur}. L'attribut fournisseur s'identifie à l'attribut id_four de la table fournisseur, mais on l'a nommé différemment ici. Il constitue une clé étrangère vers la table cible FOURNISSEUR.

- table **INGREDIENT**(id_ing : entier, nom : chaîne, numero placard : entier, stock : flottant)
Elle énumère tous les ingrédients que les confiseurs sont susceptibles d'utiliser dans leur travail, ainsi que le numéro du placard dans lequel il est conservé et le stock restant (en kilogrammes ou unité). On a choisi de les identifier par la clé id_ing. Cet entier se retrouve dans la table RECETTE et il correspond aussi à l'attribut produit de la table OFFRE.
- table **RECETTE**(id_bonbon : entier, id_ing : entier, masse : flottant). Elle indique pour chaque ingrédient la masse (en grammes) qu'il convient d'en utiliser pour la préparation de chaque bonbon dans la composition duquel il intervient³. Sa clé primaire est formée des attributs id_bonbon et id_ing. L'entier id_bonbon est une clé étrangère vers la table cible ARTICLE, il est identique à id_art. L'entier id_ing est une clé étrangère vers la table cible INGREDIENT.

On pourrait être tenté de fusionner les tables INGREDIENT et RECETTE, mais faudrait dans ce cas répéter sur chaque ligne le nom de l'ingrédient, son numéro de placard et l'état du stock, au prix d'une lourdeur inutile. Ici, ces informations ne sont écrites qu'une seule fois, même si un ingrédient donné intervient dans la fabrication de multiples friandises.

RECETTE		
<u>id_bonbon</u>	<u>id_ing</u>	masse
0	14	150
0	0	500
0	1	2
1	3	250
1	0	200
1	1	1
2	0	200
2	5	10
3	0	200
3	6	100
3	2	100
3	7	10
4	0	1500
4	8	1
4	9	1000
5	14	500
5	0	500

OFFRE		
<u>produit</u>	<u>fournisseur</u>	prix
0	1	0,83
0	2	0,81
0	3	0,93
0	4	0,87
1	2	0,16
1	3	0,17
1	4	0,19
2	0	10,40
2	1	6,80
2	3	12,90
3	0	3,90
3	3	4,10
3	2	3,70

OFFRE (suite)		
<u>produit</u>	<u>fournisseur</u>	prix
4	1	19,89
5	2	15,90
5	0	18,70
6	3	4
6	0	3,80
7	0	19,90
7	2	25,10
8	3	1,6
8	2	1,80
8	0	1,75
9	0	10,40
9	3	9,70
14	1	14,50
14	0	13,90
14	2	13,90

INGREDIENT			
<u>id_ing</u>	nom	n° placard	stock
0	sucre	1	100
1	œuf	2	1200
2	miel	1	30
3	chocolat	3	50
4	colorant rouge	3	1
5	arôme fraise	2	2
6	sirop de glucose	1	80
7	arôme de sapin	1	2
8	gousse de vanille	3	34
9	marrons	2	23
14	amande	2	17

FOURNISSEUR		
<u>id_four</u>	nom	tel
4	Metro	03 78 33 21 93
0	Patisfrance	05 63 59 05 60
3	Transgourmet	0 811 65 65 88
1	Pomona	01 37 83 22 63

FIGURE 3 – Extension de la table CONFISERIE avec quatre nouvelles tables

Dans ces tables, souligner les clés primaires et entourer les clés étrangères. Répondre aux questions suivantes.

3. Pour les œufs et les gousses de vanille, on donne le nombre.

- Quelles sont les friandises pour lesquelles on doit utiliser du sucre ?
- Quels fournisseurs permettent de préparer les bonbons que mange Jean Dujardin ?
- Quel est le coût minimal des ingrédients nécessaire à la préparation de la commande d'Angèle ?

II.4 Représentation graphique d'une BDDR

Dans les exemples précédents, on a détaillé à titre d'illustration un certain nombre de lignes de chacune de tables de la BDDR CONFISERIE. Dans la suite du cours, comme dans les énoncés de problème de concours, ce sera rarement le cas. Seul les schémas des tables seront fournis, accompagnés d'un texte explicitant la signification de chacun des attributs. Les tables utilisées par les professionnels de l'informatique peuvent comporter des millions de lignes, susceptibles de changer au fil des mises à jour des données, et il est de toute façon inenvisageable d'en afficher un nombre significatif. En pratique, on se limite donc à afficher un diagramme schématique de la base, qui reprend le nom de chaque table et de chacune de leurs colonnes. Par exemple, les tables CLIENTS et RECETTE sont réduites à ce qui suit.

CLIENTS								RECETTE		
<u>n° client</u>	Nom	Prénom	Ville	Code post.	Voie	n°	courriel	<u>id_bon</u>	<u>id_ing</u>	masse

Par commodité, il est plus courant d'énumérer les différents attributs par une liste verticale, ce qui donne la représentation de la figure (4). Le type de chaque attribut est parfois rappelé, mais les pratiques en la matière ne sont pas uniformes dans la littérature. Par paresse, je n'ai indiqué le type que pour deux des huit tables. On les relie enfin les différentes tables par des traits allant de chaque clé étrangère vers la clé primaire à laquelle elle correspond dans sa table cible. À titre d'exercice, je vous demande de le faire sur la figure ((4)).

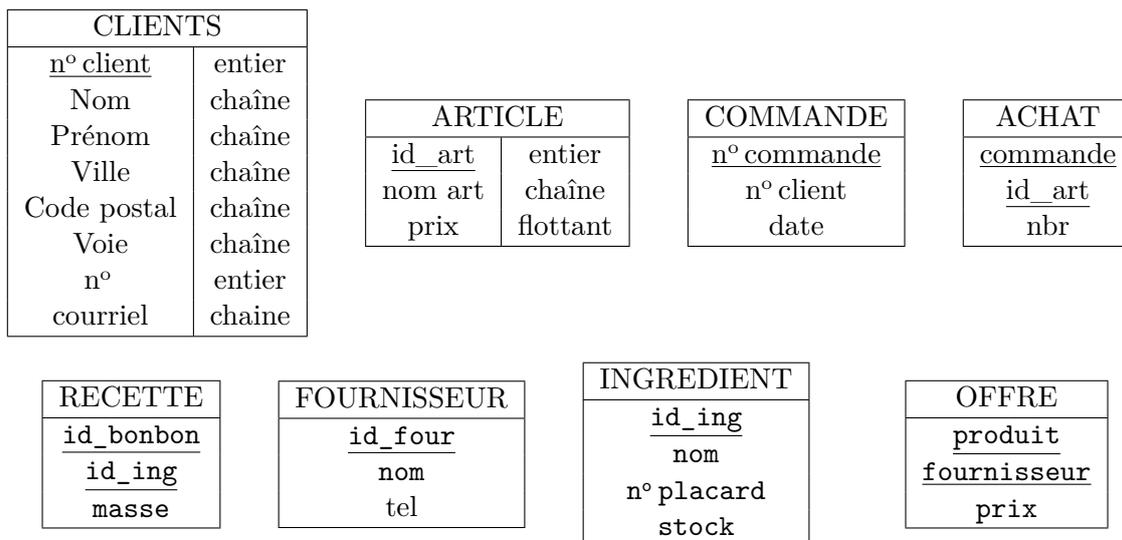


FIGURE 4 – Diagramme schématique de la table confiserie

III Modèle entité-association

III.1 Modèle conceptuel de données

Pour construire la BDDR CONFISERIE, nous nous sommes fiés à notre bon sens pour introduire une à une les 8 tables pertinentes et choisir leurs attributs. Nous espérons que le patron de la confiserie sera ainsi en mesure d’exploiter ses données afin de guider ses décisions. Cependant, il existe des situations bien plus complexes où les informations à gérer sont beaucoup plus variées, protéiformes et pour lesquelles notre approche « artisanale » serait vouée à l’échec. Certaines BDDR comportent des centaines de tables et, pour s’assurer qu’elles répondent aux besoins de leurs utilisateurs, une analyse fine s’avère indispensable. Il s’agit notamment de répondre aux questions suivantes : quelles sont les données pertinentes ? que veut-on en faire ? quelles relations existent entre ces données ? comment les organiser en tables ? quels attributs inscrire dans chaque table ?

Pour répondre méthodiquement à ces questions, il est nécessaire d’élaborer un *modèle conceptuel de données* pour le domaine d’application envisagé. Il s’agit d’une démarche de haut niveau, proche des besoins exprimés naturellement par les utilisateurs, qui décrit de manière abstraite les données d’un domaine d’application et précède tout recours à un ordinateur. Ce travail d’analyse obéit à une méthodologie rigoureuse et se fait dans le cadre du *modèle entité-association* (MEA) qui conduit à une représentation graphique par un *diagramme entité-association*.

Les contours du programme à ce sujet ne sont pas très clairs mais, à mon avis, vous aurez rarement à concevoir vous-même une base de donnée, les énoncés de sujets de concours *fournissant* le plus souvent les tables à utiliser⁴. Nous allons donc aborder succinctement quelques notions relatives au MEA. Il me semble que l’intérêt principal de ce paragraphe est de mieux comprendre la structure et le fonctionnement des bases de données. Cela peut aussi nous éclairer sur la manière de modifier une base de données préexistante afin de répondre à de nouveaux objectifs (question déjà posée aux concours). Afin de se conformer à la manière rigoureuse d’élaborer une BDD, il aurait été plus logique de faire commencer le cours par cette partie III. Mais cela aurait rendu trop pénible le début de la leçon !

III.2 Notion d’entité

Dans le cadre du modèle conceptuel EA, on utilise un vocabulaire particulier qui n’est pas exactement celui du modèle relationnel étudié dans la partie II.

- Une *entité* est un objet concret (par exemple un individu, bâtiment, une voiture) ou abstrait (par exemple une tâche, une transaction commerciale, une bataille militaire) que l’esprit peut identifier clairement. Dans une BDDR, il lui correspond un ligne (un occurrence) dans une table.
- Un *type d’entité* est un ensemble regroupant des entités similaires. Dans une BDDR, il lui correspond une table. On donne le même nom au type-entité et à la table qui lui correspond.
- Chaque entité possède des *attributs*, c’est à dire des propriétés qui la caractérise. À chaque attribut correspond une colonne d’une BDDR. Nous avons d’ailleurs déjà utilisé ce vocable « d’attribut » dans la partie I.
- Les notions concernant les *clés* introduites dans le modèle relationnel se transposent directement dans le modèle conceptuel entité-association.

Chaque notion abstraite du MEA a ainsi sa contrepartie « concrète » dans le langage des BDD relationnelles, avec souvent une appellation voisine, voire identique. À notre niveau, il n’est pas toujours utile de distinguer le niveau conceptuel et le niveau relationnel. Nous nous autoriserons donc quelques confusions, par exemple entre les mots « ligne » et « entité » d’une part, « table » et « type d’entité » d’autre part.

Puisque les types d’entité du MEA correspondent aux tables du modèle relationnel, nous les représenterons de la même manière, à savoir par un rectangle portant le nom du type d’entité dans un cartouche, et en énumérant en dessous les attributs des entités appartenant à ce type-entité. Comme il s’agit d’une de description abstraite, indépendante de son implémentation technique, on se dispense souvent de préciser le type.

Exemple de la BDD CONFISERIE

Avant même de créer les tableaux de la partie II, le confiseur réfléchit à son activité. Il identifie les type d’entités

4. Selon le programme officiel, on se limite à une description applicative des BDD en langage SQL. Il s’agit de permettre d’interroger une base présentant des données à travers plusieurs relations.

suivants : CLIENT, ARTICLE, ACHAT, PRIX, FOURNISSEUR, INGREDIENT, RECETTE, OFFRE. À chacun d'eux correspond une table portant le même nom. Angèle Van Laeke est une entité du type d'entité CLIENT ; nougat est une entité du type d'entité ARTICLE.

Autre exemple : organisation d'une entreprise

Afin de ne pas abuser des sucreries, considérons la situation suivante. Une entreprise de travaux publics est divisée en un certain nombre de départements (routes, ponts, canalisations, ligne électriques, voies ferrées, etc ...). Chaque département est dirigé par un employé qui ne peut pas en diriger d'autre. Les employés possèdent un numéro de sécurité sociale, un nom et un prénom. Chacun d'eux (sauf le directeur général) a pour supérieur hiérarchique un autre employé qui supervise son travail. Chaque employé est rattaché au département pour lequel il travaille. Cette entreprise travaille sur divers projets d'urbanisme. Chacun de ces projets possède un numéro de projet, un nom et une localisation. Chaque projet est contrôlé par un département en particulier, mais les employés des autres services peuvent tout de même intervenir dans ce projet. Par exemple, un projet d'aménagement d'un carrefour est contrôlé par le département des routes, mais les spécialistes des lignes électriques et des canalisations peuvent y contribuer.

Quels sont les types d'entités mis en jeu dans cette situation ? Je vous laisse y réfléchir et les représenter en soulignant une clé primaire pour chaque type d'entité. Donner aussi, en les imaginant, quelques exemples d'entités dans chacun des types d'entités (on ne les représente pas). On remarque que dans cette représentation, l'information contenue dans le texte ci-dessus n'a pas été exprimée dans sa totalité. Par exemple, on n'a pas spécifié que chaque employé est rattaché à un département, ni que chaque projet est piloté par un service. Ce sera l'objet du paragraphe suivant.

EMPLOYÉ
<u>numéro SS</u>
nom
prénom

DEPARTEMENT
<u>num_dep</u>
nom

PROJET
<u>num_proj</u>
nom_proj
localisation

III.3 Notion d'association

On appelle *association* un lien entre deux entités e_1 et e_2 appartenant à des types d'entités TE_1 et TE_2 . Une association est s'exprime généralement par une phrase dont le verbe désigne l'association elle-même, dont le sujet désigne e_1 et dont le complément désigne une entité de e_2 . Le cas $TE_1 = TE_2$ se rencontre parfois, on parle dans ce cas d'*association cyclique*. Toutes les associations de même nature entre les entités de TE_1 et celles de TE_2 forment un *type d'association*. Il m'arrivera parfois de dire « association » alors que je devrais dire « type d'association ». ⁵

Exemples

Alain Duchemin dirige le département des routes définit une association entre l'entité Alain Duchemin et l'entité routes.

Ali Mentation dirige le département des canalisations définit une association entre l'entité Ali Mentation et l'entité

5. Le vocabulaire n'est pas normalisé et certains auteurs appellent « association » ce que nous appelons « type d'association », et « instance d'association » ce que nous appelons « association ».

canalisations.

Ces deux associations appartiennent au type d'association **dirige**.

Nassim Eljouli supervise Kevin Lenoir et Léa Legrand supervise Pierre Lefort définissent deux associations du type d'association **supervise**. Il s'agit d'un type d'association cyclique pour le type d'entité **EMPLOYÉ**.

Dans le cas de l'entreprise de travaux publics, nous pouvons encore définir le type d'association **travaille** pour entre les type d'entité **EMPLOYÉ** et **DEPARTEMENT**, et le type d'association **contrôle** entre les types d'entité **DEPARTEMENT** et **PROJET**.

Il existe deux manière de représenter un type d'association.

- Diagramme sagittal

On représente l'ensemble contenant les entités de TE_1 et celui contenant les entités de TE_2 . On représente par une ligne fléchée chacune des associations du type d'association concerné. Quand les entités sont nombreuses, on se contente d'en représenter quelques unes dans un simple but illustratif.

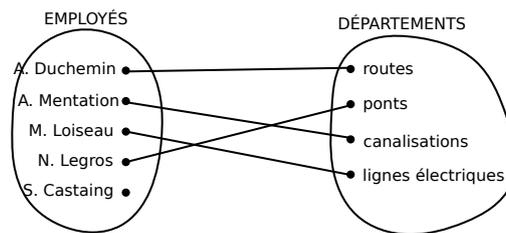


FIGURE 5 – Représentation du type d'association **dirige** par un diagramme sagittal

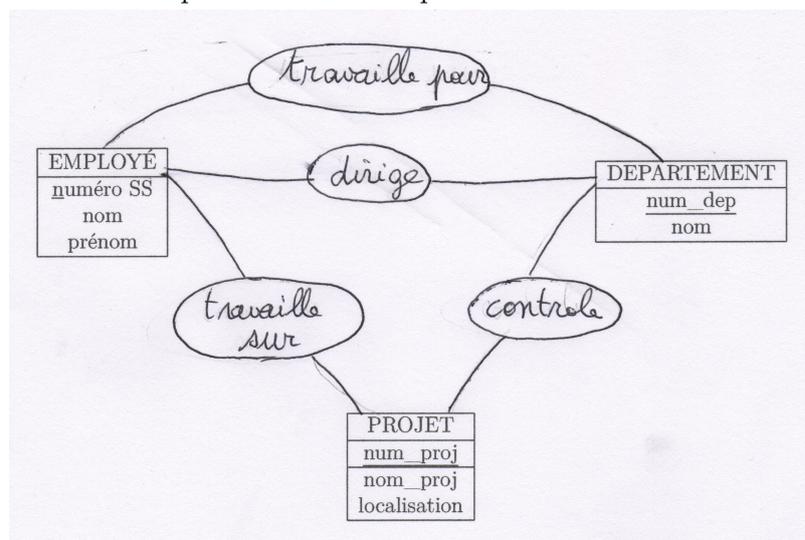
- Diagramme entité association

On représente le type d'association par une ellipse reliée par deux traits aux type d'entités TE_1 et TE_2 , eux même représentés par des rectangles comme nous l'avons vu plus haut ⁶. On omet parfois les attributs TE_1 et TE_2 pour n'écrire que le nom de ces types d'entités.



À titre d'exercice, complétons selon ce précepte le diagramme EA de la BDD ENTREPRISE.

Il manque l'association **supervise** sur mon schéma !



III.4 Classe fonctionnelle d'un type d'association

Soit A un type d'association entre deux types d'entités TE_1 et TE_2 . Sa classe fonctionnelle est définie par un couple de symboles $(N_1 : N_2)$ indiquant respectivement le nombre maximal d'associations dans lesquelles chaque élément de TE_1 (respectivement de TE_2) peuvent intervenir. Il s'agit du nombre maximal de flèches qui peuvent

6. Certains auteurs utilisent des polygones.

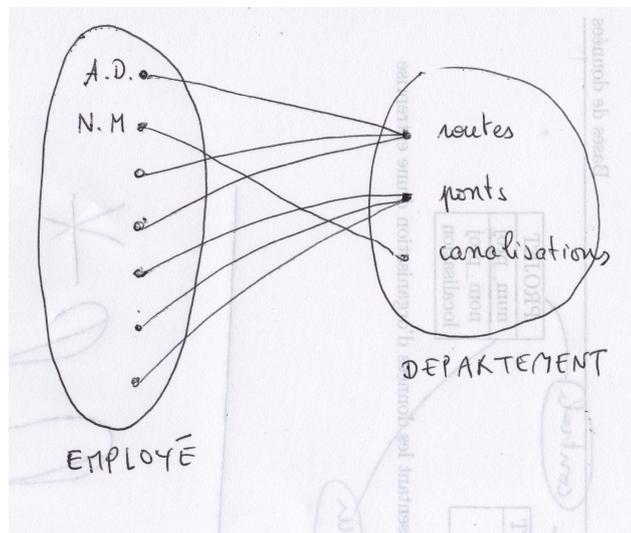
partir ou arriver sur un élément donné dans le diagramme sagittal et nous les appellerons « cardinalités » des pattes de l'association. Sur le diagramme entité association, on porte les symboles N_1 et N_2 sur les pattes qui connectent l'ellipse représentant l'association aux types d'entités TE_1 et TE_2 .



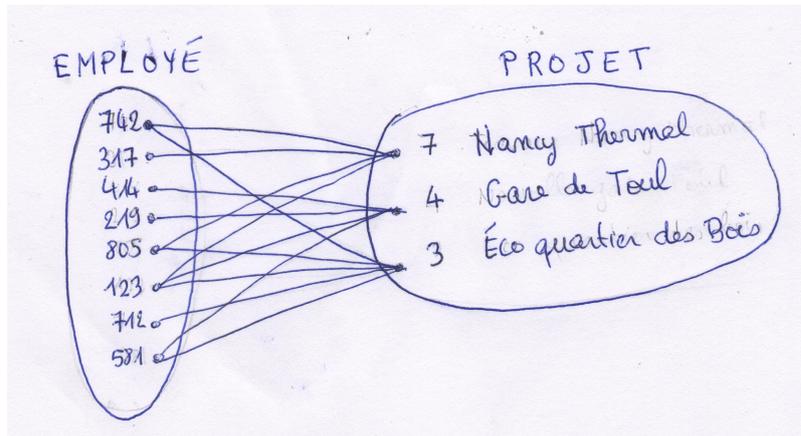
- Si chaque entité de TE_1 est associée à au plus une entité de TE_2 , on pose $N_1 = 1$. Si chaque entité de TE_1 peut être associée à un nombre quelconque d'entités de TE_2 , on écrit $N_1 = N$.
- Symétriquement, si chaque entité de TE_2 est associée à au plus une entité de TE_1 , on pose $N_2 = 1$. Si chaque entité de TE_2 est associée à un nombre quelconque d'entité de TE_1 , on écrit $N_2 = N$

Au lieu d'écrire N , on écrit parfois ∞ ou $*$ (comme dans le programme officiel) pour signifier qu'il y a pas de limite au nombre d'associations dans lesquelles une entité est susceptible de participer. En toute rigueur, on peut ainsi définir les quatre classes $(1 : 1)$, $(1 : N)$, $(N : 1)$ et $(N : N)$. En pratique, on ne distingue pas les classes $(N : 1)$ et $(1 : N)$ car on peut passer de l'une à l'autre en inversant le sens du type d'association. On écrira toujours $(1 : N)$ et le diagramme permettra de savoir à quel de côté de l'association le symbole N correspond.

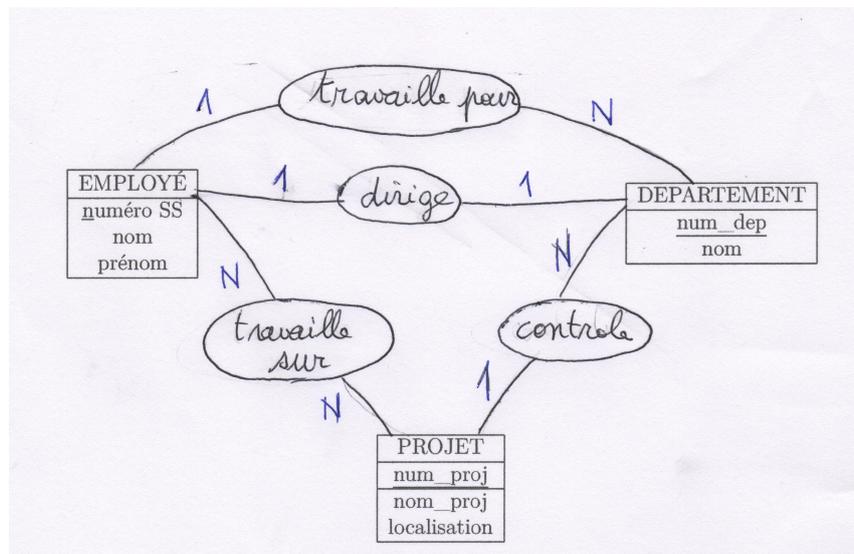
- La classe $(1 : 1)$ est aussi appelée classe « un-à-un ». Chaque élément de TE_1 est associé à au plus un élément de TE_2 , et inversement. Le type d'association **dirige** se range dans cette classe comme on le voit sur la figure (5). Chaque service a un seul directeur et chaque employé dirige au plus un service.
- La classe $(1 : N)$ est aussi appelée classe « un-à-plusieurs ». Le type d'association **travaille pour** se range dans cette classe. Pour le comprendre, observons son diagramme sagittal. Un employé donné travaille pour un seul département ($N_1 = 1$). Par contre, un département donné emploie un nombre quelconque d'employés ($N_2 = N$). Pour mieux comprendre l'expression « un-à-plusieurs », il vaut mieux lire le diagramme de droite à gauche : à département donné, on peut associer plusieurs employés. À un employé donné, on associe un seul département. Comme nous l'avons déjà dit, on ne distingue pas « de un à plusieurs » et « de plusieurs à un ».



- La classe (N, N) est aussi appelée classe « plusieurs-à-plusieurs ». Le type d'association **travaille sur** se range dans cette classe. Plusieurs employés peuvent travailler sur le même projet ($N_2 = N$) ; un employé donné peut travailler sur plusieurs projets ($N_1 = N$) en partageant son temps de travail.



Nous complétons le diagramme entité association de la BDD ENTREPRISE en indiquant 1 ou N sur chaque patte.



IV Prise en compte des associations dans une BDDR

Nous avons déjà vu plus haut que les types d'entité sont une représentation abstraite des tables, dont les lignes correspondent aux entités. L'objet de ce paragraphe est de voir comment on prend en compte les associations dans le modèle relationnel.

modèle conceptuel EA	modèle BDDR
type d'entité	table
entité	ligne
attribut	attribut
clé	clé
association	?

À FAIRE!! IL FAUT AJOUTER DES DESSINS COMME DANS HAINAUT 13.7

IV.1 Cas d'un type d'association un-à-un

Soient deux types entités TE_1 et TE_2 auxquels correspondent, dans le modèle relationnel, deux tables portant les mêmes noms. Soit \mathcal{A} un type d'association de classe un-à-un entre TE_1 et TE_2 . Pour exprimer \mathcal{A} dans le modèle relationnel, on choisit l'une des deux tables, par exemple TE_1 . On lui ajoute une (ou plusieurs) colonnes qui constituent une clé étrangère vers la table cible TE_2 . Pour chaque entité de TE_1 , elle désigne l'entité de TE_2 qui lui correspond.

Remarques :

- On peut inverser les rôles de TE_1 et TE_2 et ajouter à TE_2 une (ou plusieurs) colonnes constituant une clé étrangère vers TE_1 .

Par exemple, dans la BDD **ENTREPRISE**, on peut représenter l'association **dirige** des deux manières suivantes.

- On ajoute à l'entité **DEPARTEMENT** un attribut **numero SS dir** indiquant pour chaque département le numéro de sécurité sociale de son directeur. C'est une clé étrangère vers la table cible **EMPLOYÉ**.
- On ajoute à l'entité **EMPLOYÉ** un attribut supplémentaire **N_serv_dirigé** indiquant pour chaque employé le numéro du service qu'il dirige. L'inconvénient de cette option est que beaucoup d'employés ne sont dirigeants d'aucun département et qu'il faudrait couvrir ce cas en donnant une valeur fictive à l'attribut clé étrangère⁷. Nous préférons donc la première option.

Voici les tables obtenus avec cette manipulation. Tracer une flèche allant de la clé étrangère **numero SS dir** vers la clé primaire **numero SS**. Si on représentait le contenu de la table **DEPARTEMENT**, on verrait une nouvelle colonne.

EMPLOYÉ
<u>numero SS</u>
nom
prénom

DEPARTEMENT
<u>num_dep</u>
nom
numero SS dir

IV.2 Cas d'un type d'association un-à-plusieurs

Soient deux types entités TE_1 et TE_2 auxquels correspondent, dans le modèle relationnel, deux tables portant les mêmes noms. Soit \mathcal{A} un type d'association de classe un-à-plusieurs entre TE_1 et TE_2 . Chaque entité de TE_1 est associée à un nombre quelconque d'éléments de TE_2 , et chaque entité de TE_2 est associée à une seule entité de TE_1 . Pour exprimer \mathcal{A} dans le modèle relationnel, on ajoute à la table TE_2 (celle de cardinalité 1) une (ou plusieurs) colonnes qui constituent une clé étrangère vers la table cible TE_1 . Pour chaque entité de TE_2 , la valeur de cet attribut ajouté identifie l'entité de TE_1 qui lui correspond.

Par exemple, pour la BDD **ENTREPRISE**, on représente le type d'association **travaille pour** par une colonne **numero_dep** ajoutée à la table **EMPLOYÉ**. Elle indique, pour chaque employé, le numéro de département pour lequel il travaille et constitue une clé étrangère vers la table **DEPARTEMENT**.

De même, on représente le type d'association **CONTROLE** par une colonne **numero_dep** ajoutée à la table **projet**. Elle indique, pour chaque projet, le numéro du département qui le contrôle et constitue une clé étrangère vers la table **DEPARTEMENT**.

Avec ces ajouts, le schéma de la BDDR devient

EMPLOYÉ
<u>numero SS</u>
nom
prénom
numero_dep

DEPARTEMENT
<u>num_dep</u>
nom
numero SS dir

PROJET
<u>num_proj</u>
nom_proj
localisation
numero_dep

FIGURE 6 – Ajout de colonnes aux tables de la BDD **ENTREPRISE** pour tenir compte des types d'association **travaille pour** et **contrôle**. Tracer à la main deux flèches allant des clés étrangères introduites vers la clé primaire de la table cible.

Prenons maintenant un exemple dans la BDD **CONFISERIE**. Entre le type d'entité **COMMANDE** et le type d'entité **CLIENT** existe un type d'association de classe (1 : N), avec le N sur la patte du côté de **CLIENT** : un client peut passer plusieurs commandes mais une commande donnée est passée par un seul client. Dans la table **COMMANDE**, on a donc placé la colonne n° **client** qui indique par quel client chaque commande a été passée ; c'est une clé étrangère vers la table cible **CLIENT**.

Parler du cas cyclique (doc université strasbourg p32) et Hainaut 13.4.4 ?

7. ce qui dépasse le cadre du programme, puisque **NULL** est hors programme

IV.3 Cas d'un type d'association plusieurs-à-plusieurs

Les démarches présentées dans les deux paragraphes précédents ne sont pas très élaborées et vous auriez pu les découvrir vous-même avec un peu de réflexion. Le cas d'un type d'association plusieurs-à-plusieurs, qui figure explicitement au programme, s'avère plus délicat. Il repose sur l'astuce suivante.

Soient deux types d'entités TE_1 et TE_2 et soit \mathcal{A} un type d'association de classe un-à-plusieurs entre TE_1 et TE_2 . Pour représenter \mathcal{A} , on introduit un nouveau type d'entité TA possédant des attributs égaux aux clés primaires K_1 et K_2 de TE_1 et de TE_2 . Pour A , ces attributs pris ensemble constituent une clé primaire, et pris séparément ils sont des clés étrangères vers TE_1 et TE_2 respectivement. Les entités de TA ont pour attributs $e_1[K_1]$ et $e_2[K_2]$, avec $e_1 \in TE_1$ et $e_2 \in TE_2$. Ses « lignes » énumèrent une à une tous les instances de l'association \mathcal{A} . On établit ainsi deux nouveaux types d'association \mathcal{B} et \mathcal{B}' , de classe un-à-plusieurs, entre TE_1 et TA d'une part, entre TE_2 et TA d'autre part. L'illustration se trouve sur la figure (7).

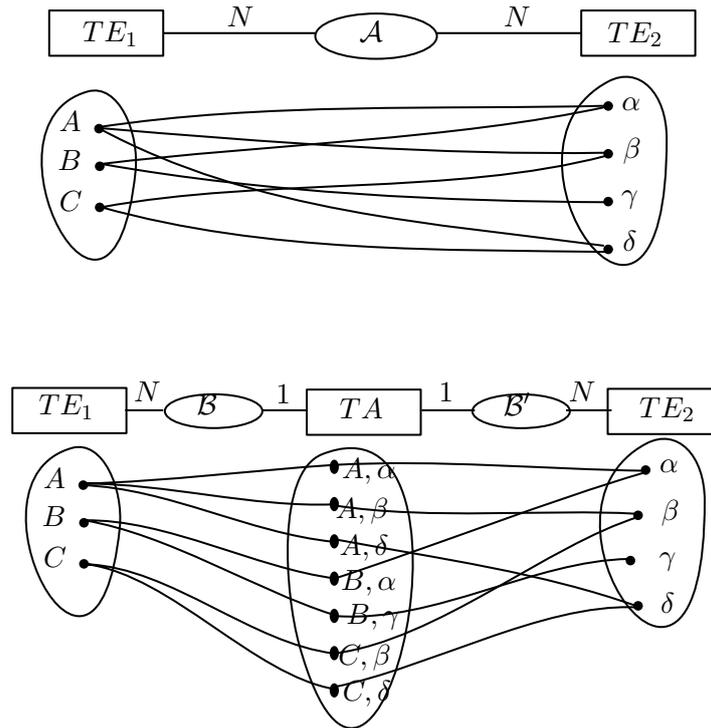


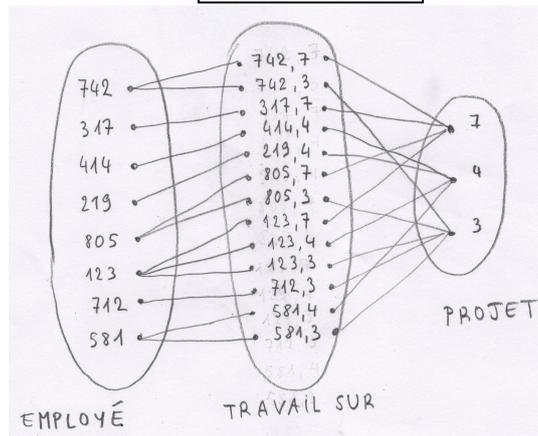
FIGURE 7 – Passage du type d'association plusieurs-à-plusieurs \mathcal{A} en deux types d'association un-à-plusieurs par introduction d'un nouveau type d'entité.

En général, \mathcal{A} est désignée par un verbe et, pour faciliter la compréhension, on nommera TA par le substantif associé. Après cette transformation de l'association \mathcal{A} en le type d'entité TA , il devient aisé de passer du MEA au modèle relationnel : TE_1 , TE_2 et TA deviennent trois tables et il suffit d'appliquer à \mathcal{B} et \mathcal{B}' , de classe un-à-plusieurs, la technique vue dans le paragraphe précédent. Le type d'association \mathcal{A} est défini par un verbe dont le sujet et le complément sont des entités de TE_1 et TE_2 . Si le verbe admet d'autres compléments intéressants⁸, il est possible de les inclure comme attributs de TA . Tout cela va s'éclaircir au travers d'exemples.

Exemple 1

Revenons au MEA de la BDD ENTREPRISE et considérons le type d'association travaille sur, de classe plusieurs-à-plusieurs. On la transforme en le type d'entité travail sur dont chaque entité est un couple constitué d'un employé et d'une tâche. On constate que du côté du type d'entité TRAVAIL SUR, chaque entité prend part, de chaque côté, à une seule association. Il n'est pas utile de donner un nom aux types d'association \mathcal{B} et \mathcal{B}' . Pour le plaisir, on pourrait les appeler EST L'OPÉRATEUR DE et EST UNE CONTRIBUTION À.

8. Ces compléments définissent des attributs de \mathcal{A}



Dans la BDDR, on a maintenant une quatrième table TRAVAIL SUR. Pour exprimer les deux associations un-à-plusieurs créées par son introduction, cette table possède deux clés étrangères : la première nommée `numero SS emp` vers la table cible EMPLOYÉ, la seconde nommée `n projet` vers la table cible PROJET. Pour satisfaire la curiosité légitime du patron de l’entreprise, on ajoute à cette table un attribut `nb heures` qui indique le nombre d’heure que chaque employé doit consacrer à chaque projet. On obtient ainsi la BDDR ci-dessous.

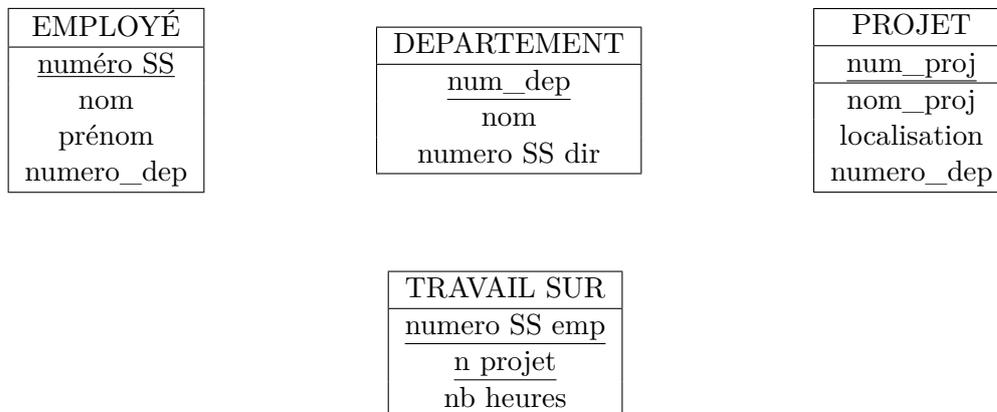


FIGURE 8 – BDDR ENTREPRISE après introduction de la table TRAVAIL SUR. Tracer flèches liant chaque clé étrangère à la clé primaire correspondante.

Exemple 2

Revenons à l’exemple de la confiserie. Si on associait directement l’entité COMMANDE à l’entité ARTICLE, l’association serait de la classe plusieurs-à-plusieurs : dans une commande donnée, on peut avoir plusieurs articles, et un article donné peut entrer dans plusieurs commandes. On évite cette situation par l’introduction de la table ACHAT, dans laquelle on retrouve les clés primaires de ARTICLE et de COMMANDE. À une ligne donné de cette table ACHAT, il correspond une unique commande et un unique achat. Ainsi, ACHAT est associé à ARTICLE et à COMMANDE par des associations de classe un-à-plusieurs.

De même, l’association « un fournisseur OFFRE un produit » est de classe plusieurs à plusieurs entre les tables INGREDIENT et FOURNISSEUR. Cette association est représentée par la table supplémentaire OFFRE, qui possède une clé étrangère référençant les ingrédients et une clé étrangère référençant les fournisseurs.

V Exercice : gestion d’un bibliothèque

Une bibliothèque gère un stock de livres que des abonnés peuvent emprunter. Chaque livre possède un titre, un auteur et un ISBN⁹. On exclut pour l’instant les ouvrages collectifs : l’auteur est unique et possède un nom et un prénom. Chaque fois qu’un nouvel ouvrage entre dans le stock, un bibliothécaire le parcourt et lui attribue quelques mots clés décrivant son contenu afin de permettre aux futurs lecteurs de procéder à des recherches informatisées. Un ouvrage donné peut exister en plusieurs exemplaires, chacun ayant sa localisation au sein de la bibliothèque. Tous les abonnés possèdent une carte portant un numéro unique et indiquant leur nom, prénom et numéro de téléphone. Chaque fois qu’un lecteur emprunte un livre, on relève la date courante et on note que le livre n’est plus en rayon jusqu’à son retour. On souhaite conserver toutes ces informations dans une BDD.

1. Construire le MEA de cette BDD (en introduisant notamment le type d’association « emprunter »).
2. Passer au au modèle relationnel et en donner une représentation sous forme de tables. Donner le schéma de relation de chacune.
3. Donner en les imaginant quelques exemples des lignes possibles pour chacune des tables.
4. Considérons dans un second temps la situation où certains livres ont été écrits par plusieurs auteurs. Comment adapter la base de données ?

Réponse sur mes notes manuscrites numérisées et placées en annexe de ce fichier pdf.

⁹ L’ISBN (International Standard Book Number) est un numéro internationalement reconnu identifiant de manière unique chaque édition de chaque livre publié dans le monde

V.1 Assurances auto

Un compagnie d'assurance automobile crée une BDD pour stocker des informations. Après discussion avec les cadres de l'entreprise, l'informaticien analyste a produit le schéma entité association de la figure 9. Les contrats de chaque client sont numérotés à partir de 1. Pour identifier sans ambiguïté un contrat, il faut donc connaître son numéro et le client qui l'a signé.

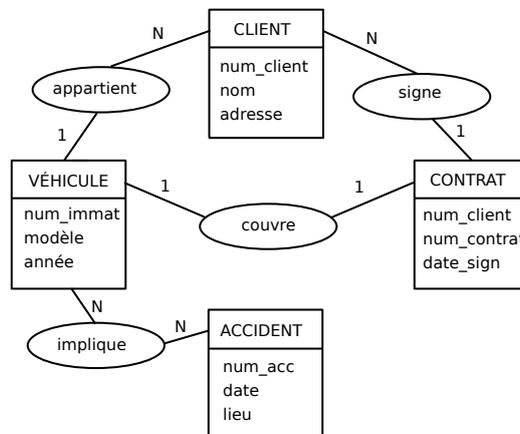


FIGURE 9 – Schéma entité association de la BDD ASSURANCE

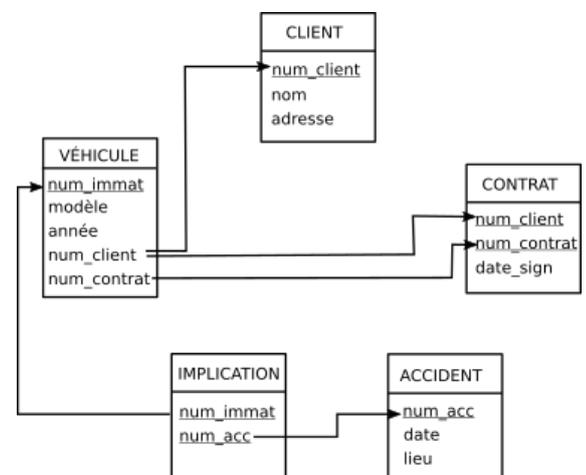
1. Exprimer ce que chaque type d'association signifie par une phrase en français comportant deux propositions (on tiendra compte des cardinalité 1 ou N). Par exemple, pour la BDD d'entreprise vue plus haut, le type d'association `contrôle`, de classe (1 :N), s'exprime par la phrase : chaque département peut contrôler plusieurs projets, un projet est contrôlé par un seul département.
2. Traduire ce schéma EA en une BDDR ; la représenter graphiquement, souligner les clés primaires et tracer des flèches allant des clés étrangères vers les clés primaires des tables ciblées.

Réponses

1. Chaque véhicule appartient à un seul client ; un client peut posséder plusieurs véhicules. Chaque client peut signer plusieurs contrats ; un contrat est signé par un seul client. Chaque contrat couvre un unique véhicule et un véhicule est couvert par un unique contrat. Un accident peut impliquer plusieurs véhicules et un véhicule peut être impliqué dans plusieurs accidents (la base de donnée correspond vraisemblablement à un historique sur plusieurs années).

2. Dans la BDDR, on crée d'abord quatre tables `CLIENT`, `VÉHICULE`, `CONTRAT`, `ACCIDENT` correspondant aux quatre types d'entités du MEA et possédant les mêmes attributs.

Pour la table `contrat`, la clé primaire est formée des deux attributs `num_client` et `num_contrat`. Le type d'association `couvre`, de classe (1 :1), est pris en compte dans la table `VEHICULE` par la clé étrangère formée des attributs `num_contrat` et `num_client` ciblant la table `contrat` (on sait ainsi quel contrat couvre quel véhicule). Le type d'association `appartient`, de classe (1 : N), est pris en compte dans la table `VEHICULE` par la clé étrangère `num_client` ciblant la table `CLIENT`. Le type d'association `SIGNE`, de classe (1 :N), est pris en compte par la clé étrangère `num_client` déjà présente dans la table `CONTRAT` (on sait ainsi quel client a signé chaque contrat). Pour traduire le type d'association plusieurs-à-plusieurs `IMPLIQUE`, on crée une table `IMPLICATION` possédant deux clés étrangères : `num_immat` vers la table cible `VEHICULE` et `num_acc` vers la table `ACCIDENT`.



Réponses à l'exercice concernant la bibliothèque

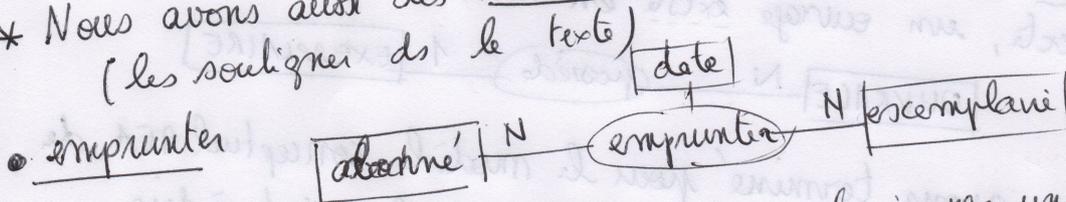
Base de données bibliothèque

- * En lisant le texte, on identifie facilement les types d'entité suivants : ouvrage, exemplaire, auteur, abonné, mot-clé. Il est important de ne pas confondre ouvrage et exemplaire. Le mot livre est un peu ambigu et il convient d'être vigilant lorsque le texte l'utilise, pour savoir s'il signifie « ouvrage » ou « exemplaire ».
- * Pour chacun de ces types d'entités, nous avons qq attributs.

OUVRAGE	ABONNÉ	MOT CLÉ	AUTEUR	EXEMPLAIRE
<u>ISBN</u>	<u>no carte</u>	<u>mot</u>	<u>nom</u>	<u>ISBN</u>
titre	nom		prenom	<u>no d'exemplaire</u>
	prenom			localisation
	tel			disponibilité

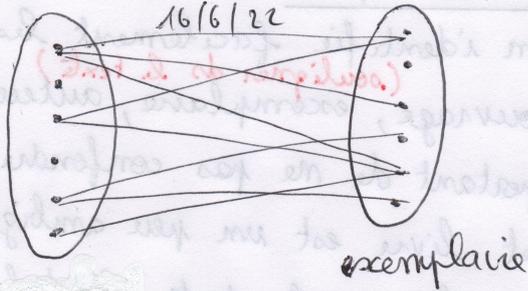
On peut déjà souligner qq clés primaires. Certaines sont formées de 2 attributs, ce qui n'est pas toujours pratique.

- * Nous avons aussi des types d'associations entre ces entités (les souligner ds le texte)



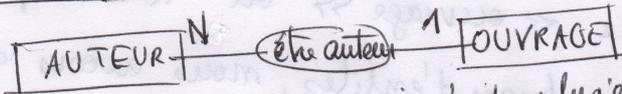
Prendre garde : on emprunte un exemplaire, pas un ouvrage ! Le texte est ambigu quant au rôle du temps. À un instant donné, un exemplaire ne peut être emprunter que par une seule personne. Mais au fil du temps, il peut être emprunté plusieurs fois, par le mêm lecteur ou par des lecteurs distincts. Nous adoptons ce second point de vue qui permet de gérer la bibliot toute l'année.

2. l'association emprunter est alors de classe (N:N).



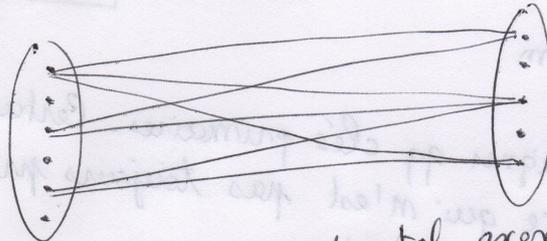
sur chaque arête, on peut spécifier la date. On a ici un attribut d'association.

• être l'auteur de

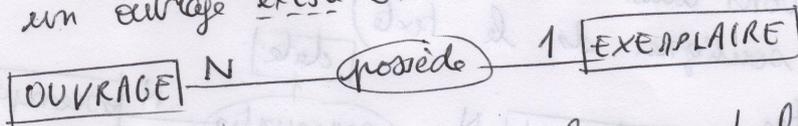


un m auteur peut avoir écrit plusieurs ouvrages

• décrire



• posséder Un ouvrage possède tel exemplaire (ds le texte, un ouvrage existe en divers exemplaires)



Nous en avons terminé pour le modèle conceptuel NEA de la BDD. Passons au modèle relationnel, c'est à dire construisons des tables.

Nous créons d'abord une table pour chaque type d'entité en recevant les attributs. Pour MOT CLE et AUTEUR, nous ajoutons un identifiant de type entier qui érite la clé primaire formée de 2 attributs. y'ai aussi introduit id-mot (un entier)

Pour exemplaire, je ne l'ai pas fait. 3

* On traduit ensuite les (1:N) être-auteur-de et posséder: avec une clé étrangère du côté de la table de cardinalité 1. Pour EXEMPLAIRE, c'est déjà fait: cette clé étrangère est ISBN.

* On traduit ensuite le type d'association (N:N) décrire en créant une nouvelle table DESCRIPTION comportant une clé étrangère vers chacune des tables OUVRAGE et MOT CLÉ (ISBN et id-mot)

* On traduit le type d'association (N:N) emprunter en créant une nouvelle table EMPRUNT. ~~La clé primaire est formée~~ Elle possède des clés étrangères: (ouvr et exemp) d'une part, vers la table cible EXEMPLAIRE, et n° carte vers la table cible lecteur.

La date de l'emprunt est un complément d'information important pour ce type d'association. On le place aussi comme attribut. Enfin, on décide d'attribuer un numéro unique à chaque emprunt: cet entier est une clé primaire commode. Attribut restitution: booléen pour savoir si le livre a été rendu.

On représente par une flèche chaque contrainte référentielle d'une clé étrangère vers une clé primaire. (cf figure)

Les schémas de table:

OUVRAGE (ISBN: entier, titre: chaîne, id-auteur: entier)

AUTEUR (id-auteur: entier, nom: chaîne, prénom: chaîne)

⁴ EXEMPLAIRE (ISBN : entier, n° ex : entier, localisation : chaîne)

ABONNÉ (n° carte : entier, nom : chaîne, prénom : chaîne, tel : entier)

MOT CLÉ (id_mot : entier, mot : chaîne)

DESCRIPTION (ISBN : entier, id_mot : entier)

EMPRUNT (id_emprunt : entier, date : chaîne, n° carte abonné : entier,ouvr : entier, exempl : entier, restitution : booléen)

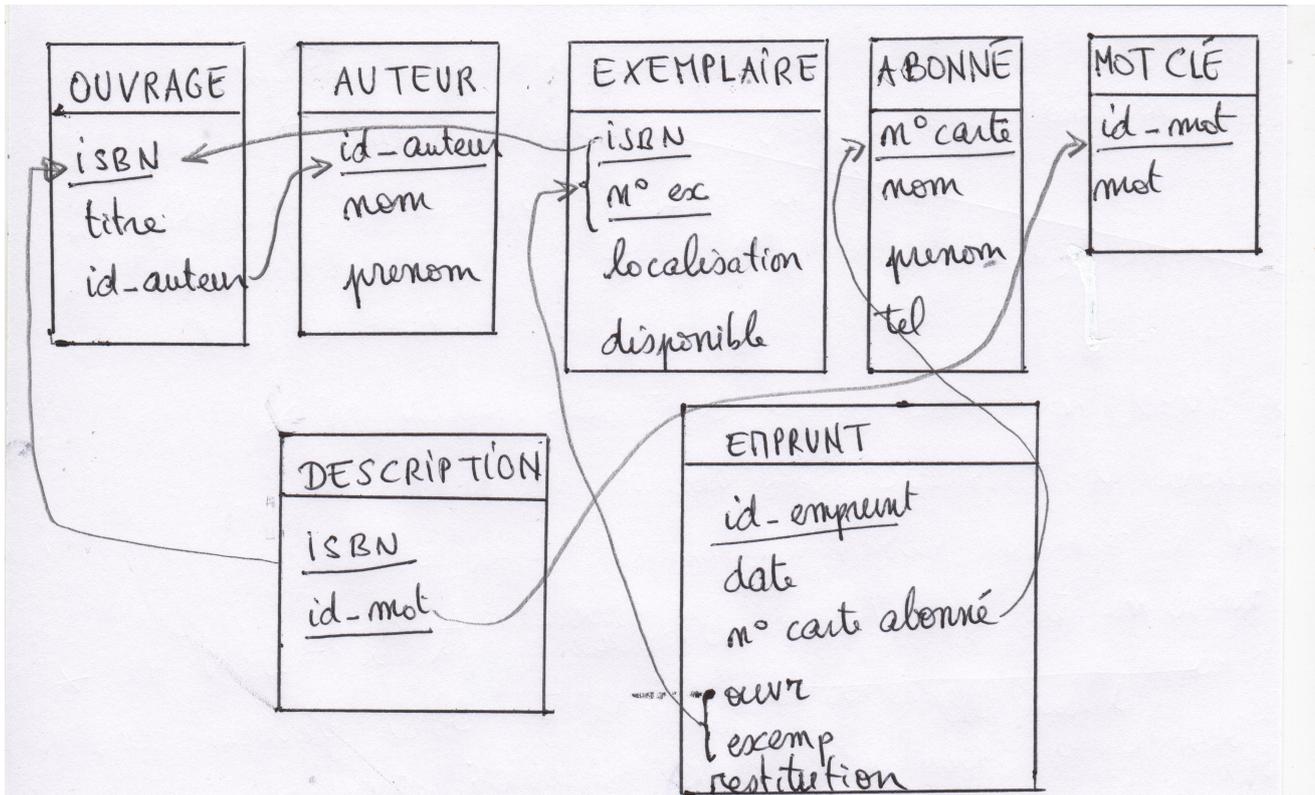
(*) Rem

Si on considère que chaque ligne peut avoir plusieurs auteurs (ouvrages collectifs), le type d'association être auteur devient de classe (N=N). On doit donc créer une table ÊTRE AUTEUR avec des clés étrangères vers OUVRAGE et AUTEUR

ÊTRE AUTEUR
ISBN
id_auteur

Donnons maintenant un aperçu du contenu de ces tables en écrivant explicitement quelques lignes. (cf fichier txt)

rem* Il peut être tentant d'attribuer un attribut booléen "disponible" à EXEMPLAIRE pour savoir si le livre a déjà été emprunté. Mais ce serait redondant avec le booléen restitution des emprunts



Essayer de refaire avec moins de croisements.

ABONNÉ			
Numéro de carte	Nom	Prénom	tél
62342	Domps	André	07 23 21 20 19
1921	Auxfraises	Charlotte	06 66 77 88 99
8234	Lebricoleur	Bob	08 33 22 91 90

AUTEUR		
1	Deiber	André
2	Hugo	Victor
3	Thorez	Jean-Paul
4	Lebrun	Kévin
5	Costill	David

OUVRAGE		
ISBN	Titre	id_auteur
978-2-7540-0915-7	Physique pour les nuls	1
978-2-253-09633-7	Les misérables	2
2-904082-75-1	Jardinage biologique	3
2-7445-0011-9	Physique du sport	4

EXEMPLAIRE		
978-2-7540-0915-7	1	premier étage
978-2-7540-0915-7	2	labo de physique
978-2-7540-0915-7	3	grenier
978-2-253-09633-7	4	CDI
978-2-253-09633-7	5	CDI
2-904082-75-1	1	armoire noire

MOT CLÉ	
0	physique
1	jardinage
2	sport
3	science
4	pauvreté
5	littérature
6	botanique
4	0
4	2
4	3

EMPRUNT					
id_emprunt	date	n° abonné	ouvr	exemp	restitution
5601	14/02/2021	1921	978-2-7540-0915-7	3	vrai
2341	14/05/2022	62342	978-2-7540-0915-7	1	faux
9382	18/07/2022	8234	2-904082-75-1	2	faux

DESCRIPTION	
978-2-7540-0915-7	0
978-2-7540-0915-7	3
2-904082-75-1	1
2-904082-75-1	6
978-2-253-09633-7	4